

Fiche technique du projet Deep Learning

Titre du projet :

Détection automatique des dommages sur les véhicules par réseaux de neurones convolutifs (CNN)

1. Objectif technique

L'objectif technique du projet est de concevoir, implémenter et comparer deux architectures de réseaux de neurones convolutifs (CNN) entièrement développées from scratch afin de détecter automatiquement des dommages visibles sur des véhicules à partir d'images.

2. Portée du projet

Le projet se limite volontairement à la tâche de détection de dommages (classification d'images ou de régions). Les fonctionnalités avancées telles que la comparaison avant/après, la génération automatique de rapports ou le déploiement industriel sont exclues du scope académique et considérées comme des évolutions futures.

3. Données

- Type de données : images RGB de véhicules
- Classes cibles :
 - véhicule sans dommage
 - véhicule avec dommage (rayure, bosse, cassure)
- Prétraitement :
 - redimensionnement des images
 - normalisation des pixels
 - augmentation de données (rotation, flip, zoom) si nécessaire

4. Architecture 1 : CNN de base (Baseline)

Description :

Cette première architecture sert de modèle de référence. Elle est volontairement simple et permet d'établir une base de performance.

Structure :

- Entrée : image RGB
- Bloc convolutionnel répété :
 - Convolution 2D
 - Fonction d'activation ReLU
 - Max Pooling
- Couches fully connected
- Couche de sortie avec Softmax / Sigmoid selon le nombre de classes

Caractéristiques :

- Architecture inspirée des blocs VGG
- Nombre modéré de couches
- Faible complexité computationnelle

5. Architecture 2 : CNN amélioré (Deep CNN)

Description :

Cette architecture vise à améliorer la capacité de détection des dommages complexes et de petite taille.

Structure :

- Entrée : image RGB
- Blocs convolutionnels plus profonds

- Introduction de skip connections implémentées manuellement
- Régularisation par Dropout
- Couches fully connected
- Couche de sortie adaptée à la tâche

Caractéristiques :

- Meilleure extraction de caractéristiques hiérarchiques
- Réduction du surapprentissage
- Architecture toujours implémentée from scratch

6. Entraînement

- Fonction de perte :
 - Binary Cross-Entropy ou Categorical Cross-Entropy
- Optimiseur :
 - SGD ou Adam
- Stratégie d'entraînement :
 - séparation train / validation / test
 - ajustement des hyperparamètres (learning rate, batch size)

7. Évaluation

- Métriques utilisées :
 - Accuracy
 - Précision
 - Rappel
 - F1-score
- Comparaison des performances entre les deux architectures

- Analyse des erreurs (faux positifs, faux négatifs)

8. Contraintes académiques

- Implémentation complète from scratch
- Aucune utilisation de modèles pré-entraînés
- Justification de chaque choix architectural
- Code clair et documenté

9. Limites

- Sensibilité aux conditions de prise de vue
- Dépendance à la qualité des annotations
- Difficulté de détection des dommages très mineurs

10. Perspectives d'évolution

- Détection et localisation précise des dommages
- Comparaison automatique avant/après location
- Génération de rapports d'état du véhicule
- Déploiement dans un environnement réel

Conclusion

Ce projet met en œuvre deux architectures CNN from scratch adaptées à une problématique réelle. Il respecte pleinement les exigences académiques et permet une analyse comparative rigoureuse des performances des modèles.